MBUS User Manual

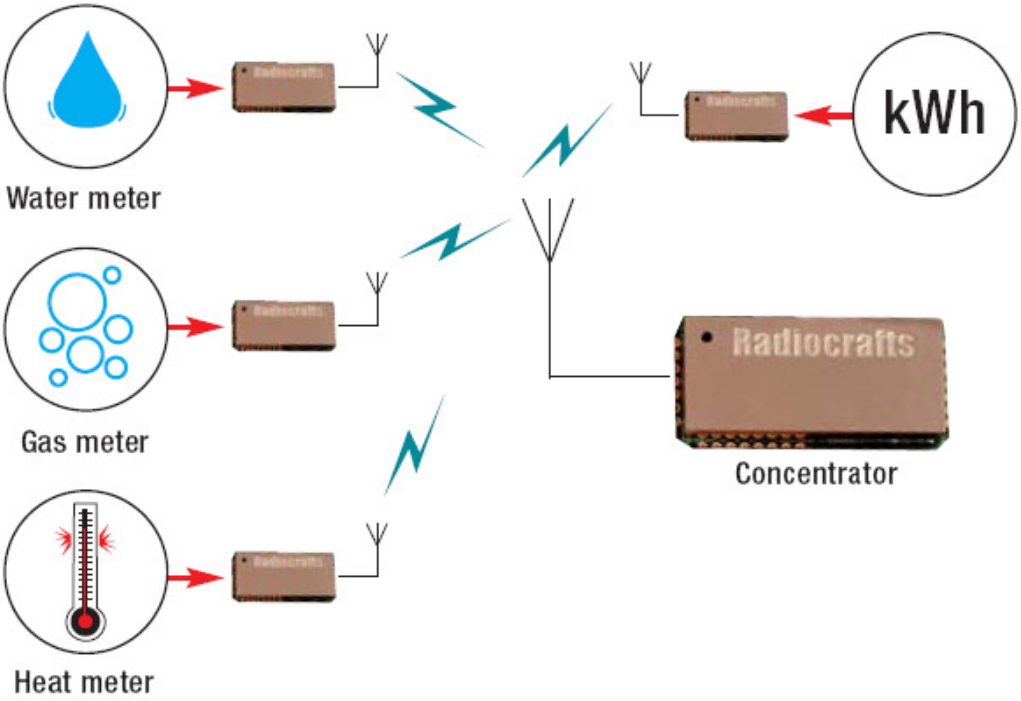


Table of contents

TABLE OF CONTENTS **2**
QUICK START **3**
MBUS FEATURE SETS **4**
OPTIONAL CUSTOM SPECIFIC VERSION **4**
IMPORTANT NOTE ON PATENT PENDING TECHNOLOGY **5**
INTRODUCTION: NETWORK TOPOLOGY **5**
WIRELESS M-BUS EMBEDDED PROTOCOL **6**
BASIC FUNCTIONALITY 6
UART INTERFACE FOR WIRELESS M-BUS PACKET HANDLING 7
FRAME FORMAT FOR TRANSMITTING DATA 8
FRAME FORMAT FOR RECEIVING DATA 8
UART INTERFACE FOR MODULE CONFIGURATION 9
UART TIMING INFORMATION 10
POWER MANAGEMENT 12
RSSI READING 12
MBUS1 DESCRIPTION **13**
MBUS1 TIMING TABLE 13
MBUS1 CONFIGURATION COMMANDS 14
MBUS1 CONFIGURATION MEMORY 16
MBUS2 DESCRIPTION **18**
MBUS2 AUTOMATIC ACKNOWLEDGE 18
MBUS2 AUTOMATIC ADDRESSING 18
MBUS2 AUTOMATIC SLEEP 18
MBUS2 INSTALLATION AND BINDING 19
MBUS2 ENCRYPTION 19
MBUS2 TIMING TABLE 22
MBUS2 CONFIGURATION COMMANDS 23
MBUS2 CONFIGURATION MEMORY 25
MBUS3 DESCRIPTION **28**
MBUS3 AUTO-MESSAGE GENERATOR 28
MBUS3 AUTOMATIC ADDRESSING 30
MBUS3 AUTOMATIC SLEEP 30
MBUS3 INSTALLATION AND BINDING 31
MBUS3 ENCRYPTION 31
MBUS3 REPEATER 32
MBUS3 TIMING TABLE 33
MBUS3 CONFIGURATION COMMANDS 34
APPENDIX: MBUS COMMAND LIST OVERVIEW **39**
APPENDIX: CONFIGURATION MEMORY FACTORY DEFAULT **40**
APPENDIX: ASCII TABLE **42**
DOCUMENT REVISION HISTORY **43**
DISCLAIMER **43**
TRADEMARKS **43**
LIFE SUPPORT POLICY **43**
CONTACT INFORMATION **43**

Quick Start

How do I transmit data?
Send your data to the RXD pin on the module. Use the UART format with settings (19200, 8, 1, N, no flow control). Up to 128 bytes are buffered in the module. The first byte of the message must contain the message length. The module will transmit the data when the whole packet is received.

How do I receive data?
Any received RF data packet with correct Wireless M-Bus format and check sums will be sent on the TXD pin. Optionally the meter address (first M-Bus block) is added to the data string. The RSSI value (received signal strength) can optionally be added to the message.

What about the antenna?
In most cases a simple quarter wavelength wire or a PCB track will do. Connect a piece of wire to the RF pin with length corresponding to the quarter of a wavelength. For space limited products, contact Radiocrafts and we will recommend the best antenna solution for your application.

How do I change the M-Bus mode, RF channel or any other parameter?
To change configurable parameters, send one byte to the module with the value 0x00 or assert the CONFIG-pin. This will take the module into configuration mode. Special commands are then used to access the configuration registers and test modes. Exit from configuration mode by sending the 'X' command. Parameters can be changed permanently and stored in non-volatile memory in the module.

MBUS feature sets

This User Manual describes the embedded protocol of the Wireless MBUS Modules from Radiocrafts. The MBUS firmware is available as different feature sets targeting specific applications. The hardware has the same size and pin-out for all frequency versions, and the different feature sets available are listed in the table below. The feature sets and the embedded functions are independent of the frequency, so this user manual is valid for all versions RC11xx-MBUSx. Detailed information on how to use the different feature sets is found in this User Manual. Additional information about the Wireless M-Bus packet structure for NTA 8130 compliance is described in Application Note 011 and is available on request.

Feature List	Feature set		
	MBUS1	MBUS2	MBUS3
General	Basic wireless M-bus functions	Added features for DSMR/NTA 8130 compliance	Added features for OMS compliance
Network role	Master or Slave	Master or Slave	Master, Slave or Repeater
Modes	S1, S2, T1, T2, R2	T1, T2	S1, S2, T1, T2
Encryption	No, must be handled externally	AES according to NTA 8130 (mode 4 and 5)	AES according to OMS (mode 4 and 5)
Installation mode	No, Must be handled externally	Yes, according to NTA 8130	Yes, according to OMS
Number of installed meters	None	Up to 8	Up to 64
Filter function	No, receives any MBUS packet. Filtering must be handled externally	Master only receives messages from installed/registered meters (optional)	Master only receives messages from installed/registered meters (optional)
Automatic acknowledge in T2	No, must be handled externally	Yes, according to NTA8130	Yes, according to OMS
Automatic message acknowledge from Master	No	No	Yes, according to OMS, supporting two-way slaves; Standard acknowledge or a predefined message from mailboxes or templates

The command set used to configure the MBUS modules are different for each feature set and an overview is found in the appendixes.

Note that this user manual also is applicable for the RCxxxxTX-MBUS. This is a TX only hardware and the RX features described in this user manual is not supported.

Optional custom specific version

As an option to the standard feature sets, a full wireless M-Bus application layer can be integrated in the module based on customer specification. In this case all the application layer protocol and timing will be handled internally by the module. See Data sheet for details.

Important note on patent pending technology

Some of the technical solutions described in this User Manual are based on patent pending technology. In particular the methods used in the MBUS3 to meet the T2 timing requirements for a master, using an address register, a flag register, an encryption key register combined with an auto-message generator for standard messages and its combination with a mailbox with pre-generated messages or templates, and a given message priority, depending on incoming messages, are subject to patenting.

Any infringements of patents and IP rights held by Radiocrafts will be prosecuted to the fullest extent.

Introduction: Network Topology

A Wireless M-Bus supported metering system normally consists of a number of heat-, gas-, water and/or electricity meters which reports their meteorological readings to a concentrator. The concentrator acts as the master in the system while the meters are slaves.

The Radiocrafts Wireless M-Bus family of modules RC11xx-MBUSx can be configured to have a role as either master or slave. The slave contains a unique address, and when sending a meter reading this address is added to the wireless message. The message from a slave does not contain any master address but the master module within range will receive the message, and based on the slave address (if the slave is installed and master configured for filtering), it will decode the message and send the data on its serial interface (TXD-pin).

In two-way communication modes, the battery operated meter (slave) will keep the receiver “on” for a short time. During this time slot the master can acknowledge the received message in order to open the communication channel (NTA 8130), or send a command (OMS) and thereby start a communication sequence. Note there is a difference in the addressing scheme between NTA 8130 and OMS: In NTA 8130 (MBUS2) the master returns an addressed acknowledgement to the slave using the address field (Link Layer Address) originally received from the slave. In OMS (MBUS3) the master sends a command with it’s own address as Link Layer Address, and the slave’s address as Application Layer Address.

MBUS3 (OMS) also allows for a one-way (unidirectional) repeater. The repeater will re-transmit all messages from slaves within range. Modules with MBUS3 features set can be configured as a repeater.

Wireless M-Bus Embedded Protocol

Basic functionality
The module offers a buffered packet radio acting as a Wireless M-Bus modem. The module contains a fully embedded protocol supporting EN13757-4:2005 modes:

- Stationary mode S (S1, S1-m, S2)
- Frequent transmit mode T (T1 and T2)
- Frequent receive mode R2

The mode is configurable by the MBUS_MODE parameter.

The required M-Bus mode is configured by setting the module in configuration mode and entering appropriate UART commands. The following modes are supported:

S1/S2-mode:
Set MBUS_MODE = 0
Set PREAMBLE_LENGTH = 0 (for short preamble) or 1 (for long preamble)
The RF channel (channel 11) and data rate (32.768 kchip/s) are set internally in the module according to the S mode, and will override any settings in the RF_CHANNEL and RF_DATA_RATE configuration registers. This setting can also be used for T2 mode slave receive and master transmit.

T1-mode:
Set MBUS_MODE = 1
The RF channel (channel 12), data rate (100 kchip/s) and preamble length are set internally in the module according to the T mode, and will override any settings in the RF_CHANNEL, RF_DATARATE and PREAMBLE_LENGTH configuration registers. This setting can also be used for T2 mode slave transmit and master receive.

T2-mode:
Set MBUS_MODE = 2
Set NETWORK_ROLE = 0 or 1
The RF channel (channel 11 or 12), data rate (32.768 or 100 kchip/s) and preamble length are set internally in the module according to the T2 mode and the selected Network Role, either being a Slave (NETWORK_ROLE = 0) or a Master (NETWORK_ROLE = 1), and change according to receive/transmit. It will override any setting in the RF_CHANNEL configuration register.

R2-mode:
Set RF_CHANNEL = 1-10
Set MBUS_MODE = 4
The data rate (4.8 kchip/s) and preamble length are set internally in the module according to the R mode.

The module supports automatic generation of the Wireless M-Bus frame, i.e.;

- Preamble (header + synchronisation)
- Adding the first block (C-field and address/manufacturing ID)
- CRC
- Postamble

The RF signal is Manchester coded or "3 out of 6" coded for increased signal integrity.

The default M-Bus mode is entered and stored in the modules' non-volatile memory (MBUS_MODE). The M-Bus mode can also be changed using the 'G' command. Using the 'G' command, the value is not stored in non-volatile memory. To do a permanent change, use the 'M' command. The 'G' command should be used for frequent change of mode, to prevent excessive writing to the flash-based non-volatile memory.

The default C-field is entered and stored in the modules' non-volatile memory (CONTROL_FIELD). The C-field can also be changed using the 'F' command. Using the 'F' command, the value is not stored in non-volatile memory. To do permanent change, use the 'M' command.

The default Manufacturer ID and unique meter Address is entered and stored in the modules' non-volatile memory. The destination address (or module address) can also be changed using the 'T' command. Using the 'T' command, the address is not stored in non-volatile memory. To do a permanent change, use the 'M' command. MBUS2 sets the destination address automatically based on the last received message.

The module has an internal buffer and transmits application data as soon as the whole packet is received based on the packet length (first byte of the application frame). The module also has a timeout feature that will empty the input buffer in case of false data packets. The default timeout is 2 seconds.Max total payload is 246 bytes, or 255 including the header in the first block.

Sleep mode can be entered via an UART command and wake-up is triggered on UART traffic (one FFh byte). Sleep mode can also be entered automatically after a transmission (configurable by SLEEP_MODE).

The module acts as a buffered packet radio, hence all data to be sent is stored in the module before they are transmitted by the RF circuitry. Likewise, when data is received they are stored in the module before they are sent to the host. This allows the communication controller to add address information, CRC and encryption during transmission, and to do error check and decryption of the received data.

The Module has an UART interface that is used for both Wireless M-Bus packet data and module configuration.

UART Interface for Wireless M-Bus packet handling
The host use the UART Interface to send and receive Wireless MBUS data. The UART packet format can be changed in the configuration mode.

When the Module receives a Wireless M-Bus packet over RF it will send the packet over the UART interface on the TXD Line. When the host MCU wants to transmit a Wireless M-Bus packet over the RF, it must send the packet through the UART Interface on the RXD line.

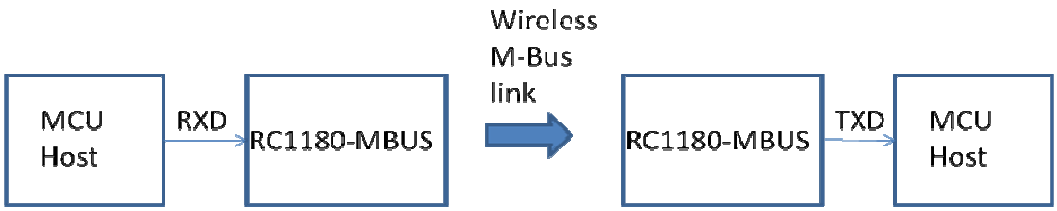


Figure 1: UART interface overview

Frame format for transmitting data
The data frame for the UART RXD pin (input for transmitting a Wireless M-Bus packet) is built like this:

Figure 2: UART interface packet transmission (RXD pin)

L is the length (not including the length byte itself), followed by the application data with the CI byte first. CI is the Control Information byte. The application data typically contains the application header, and data points with VIF and DIF codes. The application data can also be SML of DMLS.

The HEADER and C-field (and adjusted L value) is added to Wireless M-Bus packet automatically by the module before transmitting over RF and both can be changed in configuration mode.

Frame format for receiving data
The data frame for the UART TDX pin (Output for received Wireless M-Bus packets) is built like this:

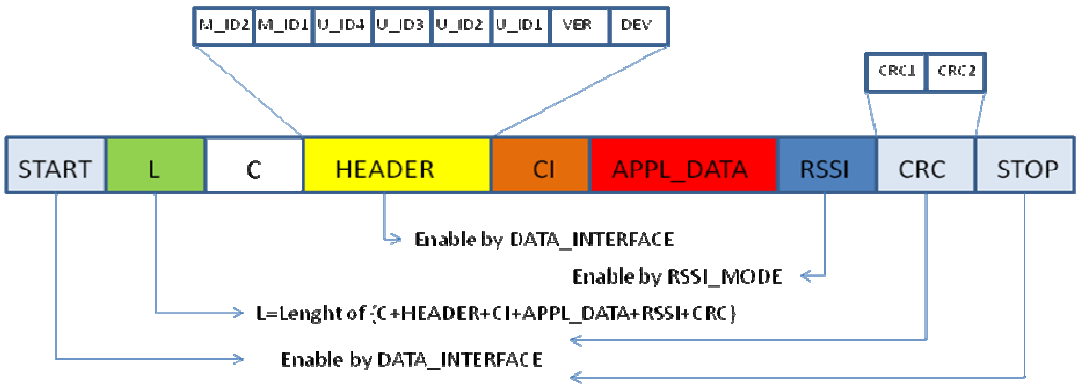


Figure 3: UART interface packet reception (TXD pin)

Data in blue and yellow are optional output parts of the UART message and can be enabled in configuration mode by the DATA_INTERFACE and RSSI_MODE configuration parameters.

L is the length byte and is always present. It does not include itself or the START/STOP bytes, but will include RSSI and CRC if enabled.

When setting DATA_INTERFACE = 1, the received HEADER will not be sent on the UART (typically used on a slave). However, to be able to notify the external application when an Acknowledgement is received (“empty” frame), a special string can be used. By setting DATA_INTERFACE = 3, the two byte string 00:E5h (i.e. L = 0) will be sent on the UART when an empty acknowledge frame is received.

Application data (CI + APPL_DATA) is always present (except when only a HEADER is transmitted).

For host applications using a UART buffer the timing information used for parsing could be lost. In this case a start and stop byte can be used. Setting DATA_INTERFACE = 4 will add a

START byte (68h) and a STOP byte (16h) to the message. This is only used for the module-to-host communication direction (TXD). Setting DATA_INTERFACE = 8 will add a two byte CRC checksum, and DATA_INTERFACE = 0Ch will add START/STOP bytes and CRC. The CRC is sent MSByte first.

The RSSI value is added when RSSI_MODE = 1.

UART Interface for Module Configuration

The configuration of the module can be changed in-circuit from the host during operation, at the time of installation of the equipment, at the manufacturing test, or even as a stand-alone module. The configuration is changed by sending commands on the UART interface after the module is set in configuration mode. The configuration mode is entered by sending 00h to the module, or by asserting the CONFIG pin (set low).

In configuration mode the module will respond by sending a '>' prompt on the TXD pin. This indicates that the module is ready to receive commands. The CONFIG pin (if used) can then be de-asserted. Note that the CONFIG pin must be de-asserted *before* the Exit command ('X') is sent to the module in order to return to normal operation.

After a command is executed, the module responds with the '>' prompt character again, indicating it is ready for a new command. Do not send a new command before the '>' prompt is received. The time required to execute a command can vary depending on the command (see the Timing Information section). There is no '>' prompt after the 'X' exit command.

The parameters that are set by dedicated configuration commands ('C', 'P' etc) take immediate effect after returning to normal operation (IDLE), but will not be stored in non-volatile memory and will be lost in case the supply power is turned off or if the module is reset. These parameters are for example the radio channel and output power.

Permanent changes of parameters can be done by writing to the configuration memory using the memory command 'M'. These are for example *default* radio channel, *default* output power and M-Bus mode, see the Configuration Memory section for details.

The flow diagram bellow illustrates how to use the UART interface to enter configuration mode, change configuration parameters and return to IDLE mode.

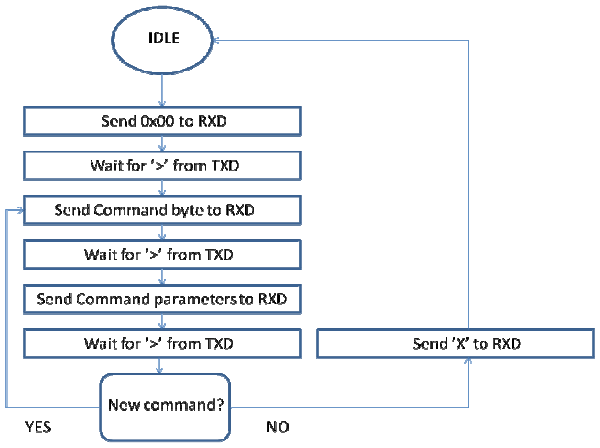


Figure 4: Configuration mode flow diagram

UART Timing Information
A UART byte consist of one start bit, 8 data bits, and one stop bit. In configuration mode a command to prompt reply will looks like this:

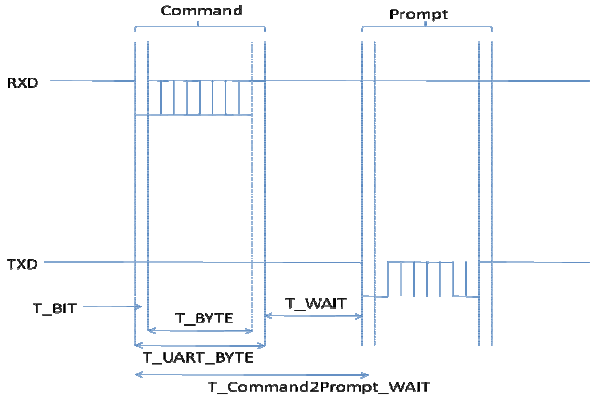


Figure 5: UART Command and prompt

The command-to-prompt wait time (T_Command2Prompt_WAIT) is different from command to command and values are shown in the timing table for each MBUS feature set.

The IDLE state is the normal state where the module both searches for preamble on the RF and wait for a character to be received on the UART. RXD is the state when receiving characters from the host filling up the internal buffer. TX state is when the data is transmitted on the air. RX state is when data is received from the air after preamble detection. TXD is the state where the received data is sent to the host on the UART.

CONFIG is the configuration mode, the state entered by sending 00h or asserting the CONFIG pin and is entered during parameter configuration, while MEMORY CONFIG is the sub-state entered by the ‘M’ command where the non-volatile configuration memory is being programmed. Note the limitation on maximum number of write cycles using the ‘M’ command, see Electrical Specifications.

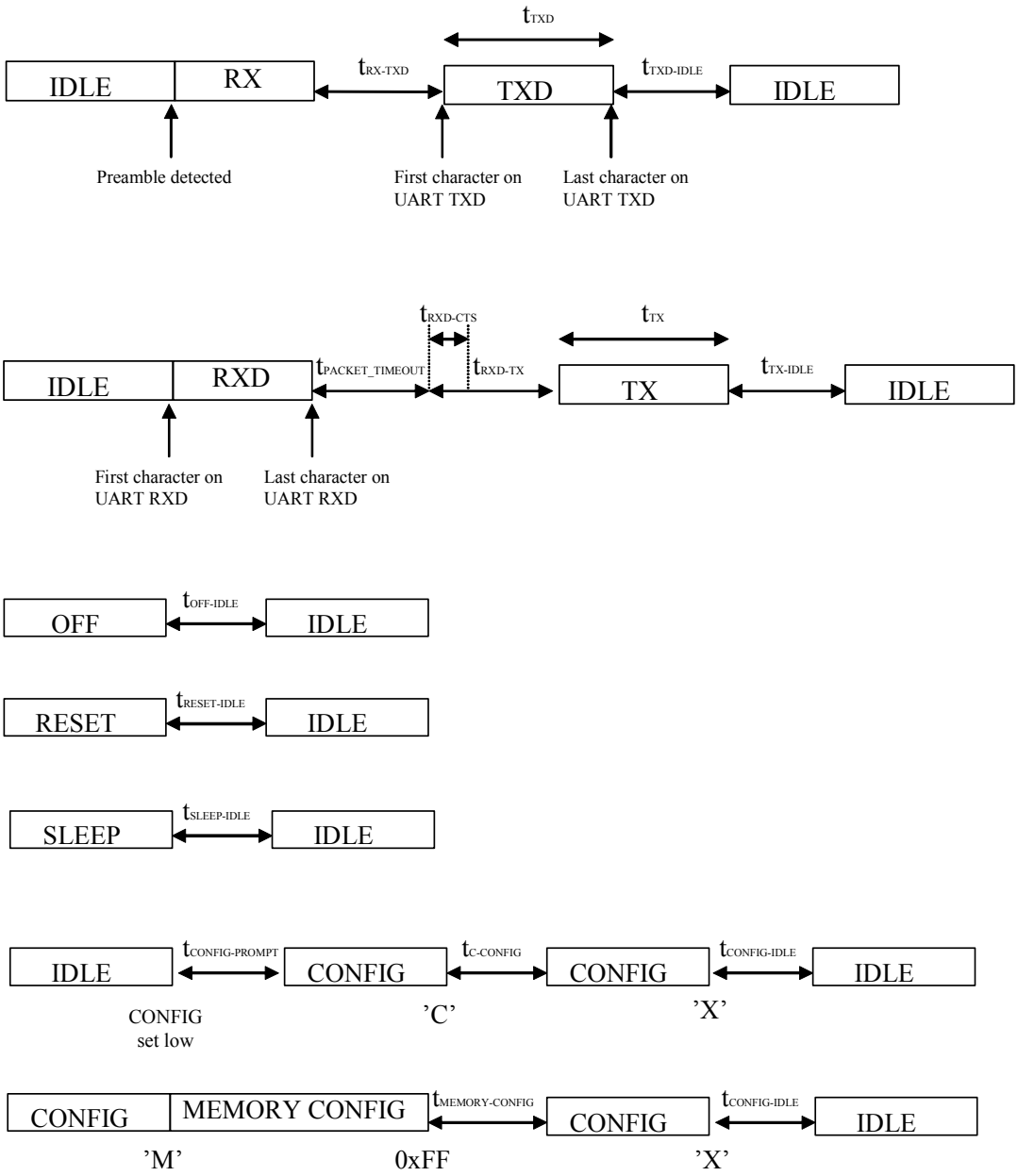


Figure 6: UART timing diagram

Timing values are shown in the timing table for each MBUS feature set.

Power Management
The module can be set in SLEEP mode in order to reduce the power consumption.

The low power SLEEP mode is manually entered by using the SLEEP command 'Z' after the module is set in configuration mode. It is also possible to configure the module to enter SLEEP automatically after a message has been transmitted (SLEEP_MODE=1). With this setup the module has to enter TX-mode (transmit a message) after power-on before entering SLEEP mode first time. In SLEEP mode the module will not receive or detect incoming data, neither from the host (UART port) nor from the air. The module is awakened from the SLEEP mode by sending the wake-up byte FFh on the UART RXD line (use a UART Baud rate > 4.8 kBd due to a maximum pulse length requirement). After the module has woken up (see Timing Information) it is ready to receive data on the UART or from the air. The SLEEP command can be used for both Master and Slave.

All configuration settings and RAM values are retained during SLEEP.

If the module is shut completely off (supply power turned off), all configuration settings in non-volatile memory is restored, but values in RAM are overwritten with default settings.

RSSI Reading
The module provide a digital Received Signal Strength Indicator (RSSI) through the 'S' command, or attached to the received messages. The RSSI value appended to a received message is the signal strength of that received packet. The RSSI value is an 8 bit character (one byte) indicating the current input signal strength or the signal strength of the received message. The signal strength can be used as an indication of fading margin, or as a carrier sense signal to avoid collisions.

The RSSI value increases with increased input signal strength in 0.5 dB steps. Input signal strength is given by (typ.):

$$P = - RSSI / 2 \text{ [dBm]}$$

MBUS1 Description

MBUS1 Timing table

The table below shows the timing information for the module when changing between different operating states. Timing symbols are according to figure 5 and 6.

Symbol	Value	Description / Note
t_{RX-TXD}	180 us	Time from last byte is received from the air until first character is sent on the UART
t_{TXD}	Min 590 us	t_{TXD} = # bytes received x 590 us/char (10 bits at 19.2 kBd + 70 us delay per character)
$t_{TXD-IDLE}$	900 us	Time from last character is sent on the UART until module is in IDLE mode (ready for RXD and RX)
$T_{RXD-CTS}$	20 us	Time from last character is received by the UART (including any timeout) until CTS is activated
t_{RXD-TX}	960 us	Time from last character is received by the UART (including any timeout) until the module sends the first byte on the air.
$T_{TX-IDLE}$	960 us	Time from last character is sent on the air until module is in IDLE mode (ready for RXD and RX)
$t_{OFF-IDLE}$	3.2 ms	
$t_{RESET-IDLE}$	3.0 ms	
$t_{SLEEP-IDLE}$	1.3 ms	
$t_{CONFIG-PROMPT}$	60 us	Time from 00h / CONFIG pin is set low until prompt ('>')
$T_{C-CONFIG}$	1.1 ms	Delay after channel-byte is sent until prompt (">").(For other volatile memory commands there is no delay but immediate prompt)
$T_{G-CONFIG}$	1.1 ms	Delay after new M-Bus mode-byte is sent until prompt ('>'). (For other volatile memory commands there is no delay but immediate prompt)
T_{WAIT}	1.55 ms (M command) 24 us (all other commands)	Delay from stop bit of the command byte to start bit of the prompt reply. See figure 5 for details.
$t_{MEMORY-CONFIG}$	31 ms	In this period the internal flash (non-volatile memory) is programmed. *Do not reset, turn the module off, or allow any power supply dips in this period as it may cause permanent error in the Flash configuration memory. After the last command parameter byte the host should wait for the '>' prompt before any further action is done to ensure correct re-configuration.*
$T_{CONFIG-IDLE}$	1.1 ms	End of 'X' to IDLE
t_{TX}	3.6 ms	TX time for T1 mode when Length=1 on the UART. Preamble, sync, CRC and MBUS address field added internally.

MBUS1 Configuration Commands
A list of commands is shown in the table below. Commands must be sent as ASCII characters or their corresponding binary value. All arguments must be sent as binary values to the module (not as ASCII representation for hex or decimal).

Parameter	Command	Argument in hex (decimal)	Note
Channel	'C' – 0x43	0x01-0x0C (1-10) Apply for R mode only	Data is stored in volatile memory only.
C-field	'F' – 0x46	0x00-0xFF (0-255)	Data is stored in volatile memory only.
M-Bus mode	'G' – 0x47	0x00-0x04 (0-4) 0: S 1: T1 2: T2 3: Reserved 4: R	Data is stored in volatile memory only.
Memory configuration	'M' – 0x4D	(Address, Data): see list of parameters below. 0xFF exits memory configuration.	Used to enter memory configuration menu. Parameters changed are stored in non-volatile memory.
Output power	'P' – 0x50	0x01-0x05 (1-5)	Data is stored in volatile memory only.
Quality Indicator	'Q' – 0x51	Returns one byte indicating the signal quality	Based on bit errors in preamble and synch word
Signal Strength (RSSI)	'S' – 0x53	Returns one byte indicating the signal strength of a detected signal or a valid packet.	If a valid packet has been received when in configuration mode, it will return the RSSI of the last received packet.
Destination / module address	'T' – 0x54	8 bytes; MAN_ID2 (Second manufacturer code), MAN_ID1 (First manufacturer code), ID4, ID3, ID2, ID1, VER (Version), DEV (Device Type),	Data is stored in volatile memory only.
Exit command	'X' – 0x58	(none)	Exit to normal operation mode. All changes of parameters take effect.
Sleep mode	'Z' – 0x5A	(none)	Exit sleep mode by sending 0xFF on UART RXD pin
Test mode 0	'0' – 0x30	(none)	List all configuration memory parameters
Test mode 1	'1' – 0x31	(none)	TX carrier
Test mode 2	'2' – 0x32	(none)	TX modulated signal PN9 sequence
Test mode 3	'3' – 0x33	(none)	TX Off, RX mode

Note: ASCII characters are written as 'X', hexadecimal numbers are written like 0x00, and decimal numbers are written like 10 throughout the text. A table of ASCII characters and their respective hex and decimal values are found in the Appendix.

Any invalid command will be ignored and the '>' prompt will be re-sent.

If Test mode 1 or 2 is used, it is important to enter Test mode 3 before exiting the configuration mode ('X') in order to ensure proper operation in normal mode.

Example:
To select RF channel 3, send the follow sequence after asserting the CONFIG line and the '>' prompt is received:

Command	**Hex**	**Response**	**Comment/Note**
Enter	0x00	'>'	Or assert CONFIG pin De-assert CONFIG after '>' prompt
'C'	0x43	'>'	
3	0x03	'>'	Wait for '>' prompt
[A new command could be issued here]			
'X'	0x58	(none)	Module returns to IDLE state

Note that the CONFIG line must be de-asserted after the first '>' prompt was received, but before the 'X' command.

MBUS1 Configuration Memory

The table below shows the complete list of configurable parameters stored in non-volatile memory. These values can be changed using the 'M' command. All addresses and arguments must be sent as binary values to the module (not as ASCII representation for hex or decimal).

Parameter	Description	Address hex	Argument dec	Factory setting hex (dec)	Comment
Radio configuration					
RF_CHANNEL	Default RF channel for R mode only	0x00	1-10	0x01 (1)	See data sheet for channel frequencies. Only used for R mode.
RF_POWER	Default RF output power	0x01	1-5	0x05 (5)	See data sheet for output power levels.
MBUS_MODE	M-Bus mode	0x03	0-4 0: S 1: T1 2: T2 3: Reserved 4: R	0x01 (1)	Use 'G' command to change value in volatile memory only
SLEEP_MODE	Sleep mode	0x04	0: Disable Sleep 1: Enable Sleep	0x00 (0)	When enabled the module enter Sleep mode after transmission
RSSI_MODE	Append RSSI to received data	0x05	0:Disabled 1: Enabled	0x00 (0)	When enabled the RSSI value is appended to the received data
Radio packet configuration					
PREAMBLE_ LENGTH	Short of long preamble in S mode only	0x0A	0x00 (0): Short 0x01 (1): Long	0x00 (0)	Preamble (header) length, apply for S mode only
Medium access, addressing and network management					
NETWORK_ROLE		0x12	0x00 (0): Slave/Meter 0x01 (1): Master/Concentrator	0x00 (1)	
M_ID1	Manufacturer ID, first byte	0x19	0x00-0xFF (0-255)	0x0C (12)	
M_ID2	Manufacturer ID, second byte	0x1A	0x00-0xFF (0-255)	0xAE (174)	
U_ID1	Unique ID, first byte	0x1B	0x00-0xFF (0-255)	0x12 (18)	
U_ID2	Unique ID, second byte	0x1C	0x00-0xFF (0-255)	0x34 (52)	
U_ID3	Unique ID, third byte	0x1D	0x00-0xFF (0-255)	0x56 (86)	
U_ID4	Unique ID, forth byte	0x1E	0x00-0xFF (0-255)	0x78 (120)	
VER	Version	0x1F	0x00-0xFF (0-255)	0x01 (1)	
DEV	Device	0x20	0x00-0xFF (0-255)	0x07 (7)	
Data and configuration interface, UART Serial Port					
UART_BAUD_RATE	Baud rate	0x30	0x00: Not used 0x01: 2400 0x02: 4800 0x03: 9600 0x04. 14400 0x05: 19200 0x06: 28800 0x07: 38400 0x08: 57600 0x09: 76800 0x0A: 115200 0x0B: 230400	0x05 (5)	BE CAREFUL IF CHANGING AS HOST MAY LOOSE CONTACT WITH MODULE! Does not take effect until module is re-booted / reset.
UART_FLOW_CTRL	UART flow	0x35	0: None	0x00 (0)	

	control		1:CTS only 3:CTS/RTS 4:RXTX(RS485)		
DATA_INTERFACE	Data interface	0x36	0x00: MBUS packet with ID and address 0x01: Application data only 0x02: Reserved 0x03: Application data only with ack (00:3Eh) 0x04: Add start/stop byte 0x08: Add CRC 0x0C: Add start/stop byte and CRC	0x00 (0)	Sets receiver data format. First byte is always packet length (except when using start byte)
CONTROL_FIELD	C-field	0x3B	0x00-0xFF (0-255)	0x44 (68)	Use 'F' command to change value in volatile memory only
PART_NUMBER		0x3C-0x48		RCxxxx-MBUS1	
HW_REV_NO		0x4A-0x4D		x.yz	x, y and z; Any number 0d-9d
FW_REV_NO		0x4F-0x5		x.yz	x, y and z; Any number 0d-9d
Exit from memory configuration		0xFF	No argument should be sent		To exit from command mode the 'X' command must be sent after '>' is received.

To make permanent changes to default values and other parameters, the Memory Configuration command 'M' is used. This command should be followed by pairs of byte being the memory address and the new value to be stored at that address. In order to exit the Memory Configuration mode, the 'address' 0xFF must be sent, but without any data argument. Then wait for the '>' prompt while the internal memory is re-programmed (see Timing Information for typical delay). To completely exit from command mode, the normal exit command 'X' must be sent.

Example:
To change the MAN_ID (at address 0x19 and 0x1A) and set it to (100,200) (0x64, 0xC8), send the following sequence:

Command	**Hex**	**Response**	**Comment/Note**
Enter	0x00	'>'	Or assert CONFIG pin De-assert CONFIG after '>' prompt
'M'	0x4D	'>'	Module ready to receive address
0x19	0x19	(none)	
100	0x64	(none)	
0x1A	0x1A	(none)	
200	0xC8	(none)	
[new address could be sent here]			
[new value could be sent here]			
0xFF	0xFF	'>'	Wait for '>' prompt
'X'	0x58	(none)	Module returns to IDLE state

Test mode 0 ('0' command) can be used to list all parameters stored in non-volatile memory. This command can be used to verify and check the module configuration.

MBUS2 Description

MBUS2 Automatic Acknowledge

The Master must reply with an acknowledge message within 3 ms after a received Access Demand, if further communication shall take place. To meet this timing requirement, the module has built-in automatic acknowledge support. Use the Acknowledge flag (set using the A –command) to indicate which slave shall be acknowledged at the next access. The flag is automatically cleared but can also be cleared manually with the A-command.

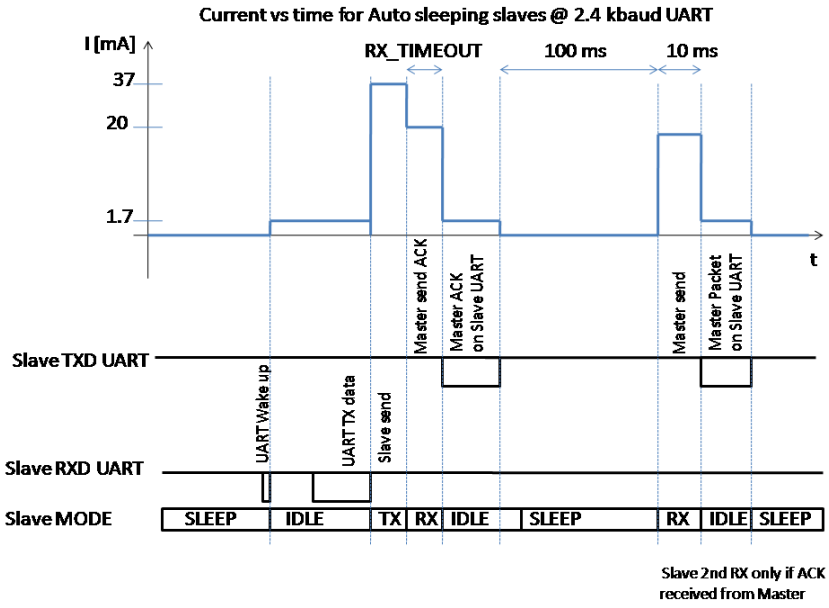
MBUS2 Automatic Addressing

When the Master receives a message from a slave, this slave HEADER will be used as the HEADER for the next transmission from the master. If, for example during installation, messages are received from several meters, the last message received will be the default address. This feature can be overridden by using the 'T' command.

MBUS2 Automatic Sleep

The Slave has special support for automatic sleep after data transmission. If automatic SLEEP is enabled (SLEEP_MODE = 1), the module will automatically go to sleep in a configurable time after data transmission, if acknowledge is not received. The receiver timeout after slave TX is configured by RX_TIMEOUT. If acknowledge is received after the slave transmission, the slave goes to sleep for 100 ms before it goes into RX for 10 ms waiting for a new Master message. This reduces the burden of the slave host controller when meeting NTA 8130 (v.3.0) data exchange timing.

A '0xFF' byte will wake up a sleeping Slave into a low power idle mode (IDLE). The Slave will only accept UART input (RXD UART) in this IDLE mode to save current. The current saving depends on UART rate and gives less current consumption for low UART rates (but data transfer takes longer time). The figure below illustrates current vs time for a Slave configured for auto sleep at 2.4 kBaud UART rate when receiving acknowledge from Master after the first transmission.



MBUS2 Installation and Binding
The module (Master) can be set in Installation Mode using the "I" command. When the module is in Installation Mode it will accept all Access Demand Install messages (C-field is 46h in T1 and 06h in T2).

Slaves can be bound to a Master by registering their addresses in the Address Register. This is done by using the "B" (Bind) command followed by a register number (1-8) and an 8 bytes slave address. Thus, a maximum of 8 meters can be bound to one Master for the MBUS2 feature set (meeting NTA 8130 request for minimum 4 meters).

Note; the host must know which registers are used and which are free at any time.

MBUS2 Encryption
The module supports AES-128 encryption. When a slave is registered into the masters address register, the master host should request a new encryption key from the utility data base. The new key is specific for each slave and related to the slave equipment ID or unique address. The new key should be provided in two versions; plain and encrypted using the slave's default key. That is, the utility needs to keep a register with default keys linked to each meter.

The master host should send the new encrypted key to the slave. And the slave host should configure this new key into the module using the 'K' command, followed by 16 bytes (the encrypted key). The slave module will automatically de-crypt the new key using its default key.

The master host should then send the new (plain) key to the master module using the 'K' command, followed by the register location number, and the 16 bytes (the new key). The register location number must correspond to the address register location for that slave.

The ENCRYPT_FLAG and DECRYPT_FLAG parameters are used to enable / disable the encryption when transmitting and receiving messages. The 8 bit values are interpreted as bit maps corresponding to the 8 address registers, LSB being register 1.

For a message to be encrypted, the encryption flag for the particular slave must be set, and the CI-field and Signature field sent to the module must be according to the standard for encryption to take place. The module will do byte stuffing if required to get a full 16 byte encryption block. Only CI-fields 0x5A, 0x5B and 0x72 allows encryption. The Signature field must be encryption mode 0x04 or 0x05 according to NTA 8130.

When using Signature field 0x05, the application must add the two encryption verification bytes (0x2F) after the header. The Initialization Vector for the encryption is extracted from the long header (for CI-fields 0x5B and 0x72). For the short header (CI-field 0x5A) the Initialization Vector is partly from the MAC header (destination address) and the short application header.

The Access Counter byte in the application header is used by the encryption, and the host application must increment the counter in order to avoid repetitive messages.

For a message to be decrypted the decryption flag for the particular slave must be set, and the CI-field and Signature field must be according to the standard for encryption to take place.

In the slave, only LSB is used as a flag for encryption/decryption.

The ‘D’ and ‘E’ commands are used to set decryption and encryption flags without storing in non-volatile memory. The value following the D and E commands is interpreted as bit maps corresponding to the 8 address registers.

To test the encryption feature you need to have a valid key set for the master and the slave. In addition you need to send a valid UART frame into the module in order for the internal encryption and decryption feature to be activated on this message. The RCTools PC software from Radiocrafts (MBUS_CCT and MBUS_DEMO) can be used to configure key sets and send and receive encrypted messages.

Example of a key set:
Master Key: 0x00 0x11 0x22 0x33 0x44 0x55 0x66 0x77 0x88 0x99 0xAA 0xBB 0xCC 0xDD 0xEE 0xFF
Slave Key: 0x0A 0x90 0xE5 0xB7 0x4D 0x28 0x07 0xA6 0x51 0xF6 0x9A 0xC0 0x89 0x6A 0x09 0xF6
Use factory default for Init vector and Default key in the configuration memory.

Example of UART RXD frames that enable encryption:

Test packet A: No filling byte
Slave TX message: C=6, L=2D, CI=72,
Data=78563412AE07010701002004 2F2FAABBCCDDEEFFAABBCCDDEEFFAABBCCDDEEFFAABBCCDDEEFFAABBCCDDEEFF

Slave TX message: C=6, L=25, CI=5A,
Data=020020042F2FAABBCCDDEEFFAABBCCDDEEFFAABBCCDDEEFFAABBCCDDEEFFAABBCCDDEEFF

Slave TX message: C=6, L=2D, CI=5B,
Data=78563412AE07010703002004 2F2FAABBCCDDEEFFAABBCCDDEEFFAABBCCDDEEFFAABBCCDDEEFFAABBCCDDEEFF

Test packet B: Adding filling bytes
Slave TX message: C=6, L=1E, CI=72,
Data=78563412AE07010704001104 2F2FAABBCCDDEEFFAABBCCDDEEFFAABBCC

Slave TX message: C=6, L=17, CI=5A,
Data=0500120 42F2FAABBCCDDEEFFAABBCCDDEEFFAABBCCDD

Slave TX message: C=6, L=2E, CI=5B,
Data=78563412AE07010706002104 2F2FAABBCCDDEEFFAABBCCDDEEFFAABBCCDDEEFFAABBCCDDEEFFAABBCCDDEEFFAA

Test packet C: Adding filling bytes and un-encrypted bytes at the end of the packet.
Slave TX message: C=6, L=22, CI=72,
Data=78563412AE07010707001104 2F2FAABBCCDDEEFFAABBCCDDEEFFAABBCCAAAAAAAA

Slave TX message: C=6, L=1B, CI=5A,
Data=0800120 42F2FAABBCCDDEEFFAABBCCDDEEFFAABBCCDDAAAAAAAA

Slave TX message: C=6, L=2B, CI=5B,
Data=78563412AE07010709001A04 2F2FAABBCCDDEEFFAABBCCDDEEFFAABBCCDDEEFFAABBCCDDEEFFAAAAAAAA

Green: Number of encrypted bytes including the two 0x2F bytes in the start of the encryption part
Read: Second block Header (un-encrypted)
Blue: Signature field. To enable encryption this field must be 04 (NTA3130 encryption) or 05 (NTA8130 encryption harmonized to OMS).
Purple: Encrypted part of UART message. Internal encryption and not encrypted on UART
Black: Un-encrypted part of message

Slave step-by-step guide to send an encrypted message to a Master:
1. Enter configuration mode
2. Send the slave key to the module using the ‘K’ command and ‘Slave Key’ as command parameter.
3. Enable Encryption using the ‘E’ command and ‘0x01’ as command parameter
4. Leave configuration mode using the ‘X’ command

5. Send one of the valid test packets above.

This packet will now be encrypted over the RF link.

Master step-by-step guide to receive an encrypted message from a Slave:
1. Enter configuration mode
2. Install the Slave to this Master using the ‘B’ and ‘Address Register’ as command parameter followed by the slave address. Address register is 1-8.
2. Send the Master key to module using the ‘K’ command and ‘Address Register’ + ‘Master Key’ as command parameters.
3. Enable Decryption using the ‘D’ command and ‘Address Register’ as command parameter.
4. Leave configuration mode using the ‘X’ command

An encrypted packet from the installed slave will now be decrypted before it is provided on the UART TXD.

Note that you also have the option to use the M command to permanently set the Encryption/Decryption flag in the configuration non-volatile memory, instead of using the ‘D’ and ‘E’ commands. Encryption / decryption will only take place when the signature field indicate mode 0x04 or 0x05. If encryption mode 0x00 is used, the message will not be encrypted / decrypted even if the flags are set.

MBUS2 Timing table
The table below shows the timing information for the module when changing between different operating states. Timing symbol is according to figure 5 and 6.

Symbol	Value	Description / Note
t_{RX-TXD}	180 us	Time from last byte is received from the air until first character is sent on the UART
t_{TXD}	Min 590 us	t_{TXD} = # bytes received x 590 us/char (10 bits at 19.2 kBd + 70 us delay per character)
$t_{TXD-IDLE}$	900 us	Time from last character is sent on the UART until module is in IDLE mode (ready for RXD and RX)
$T_{RXD-CTS}$	20 us	Time from last character is received by the UART (including any timeout) until CTS is activated
t_{RXD-TX}	960 us	Time from last character is received by the UART (including any timeout) until the module sends the first byte on the air.
$T_{TX-IDLE}$	960 us	Time from last character is sent on the air until module is in IDLE mode (ready for RXD and RX)
$t_{OFF-IDLE}$	3.2 ms	
$t_{RESET-IDLE}$	3.0 ms	
$t_{SLEEP-IDLE}$	1.3 ms	
$t_{CONFIG-PROMPT}$	60 us	Time from 00h / CONFIG pin is set low until prompt (“>”)
$T_{G-CONFIG}$	1.1 ms	Delay after channel-byte is sent until prompt (“>”).(For other volatile memory commands there is no delay but immediate prompt)
$T_{G-CONFIG}$	1.1 ms	Delay after new M-Bus mode-byte is sent until prompt (“>”).(For other volatile memory commands there is no delay but immediate prompt)
T_{WAIT}	1.55 ms (B, K and M command) 24 us (all other commands)	Delay from stop bit of the command byte to start bit of the prompt reply. See figure 5 for details.
$T_{MEMORY-CONFIG}$	31 ms	In this period the internal flash (non-volatile memory) is programmed. *Do not reset, turn the module off, or allow any power supply dips in this period as it may cause permanent error in the Flash configuration memory. After the last command parameter byte the host should wait for the ‘>’ prompt before any further action is done to ensure correct re-configuration.*
$T_{CONFIG-IDLE}$	1.1 ms	End of ‘X’ to IDLE
t_{TX}	3.6 ms	TX time for T1 mode when Length=1 on the UART. Preamble, sync, CRC and MBUS address field added internally. Depends on M-Bus mode (T, S, R) and L

MBUS2 Configuration Commands

A list of commands is shown in the table below. Commands must be sent as ASCII characters or their corresponding binary value. All arguments must be sent as binary values to the module (not as ASCII representation for hex or decimal).

Parameter	Command	Argument in hex (decimal)	Note
Acknowledge	'A' – 0x41	One byte bitmap, address register 1 is LSB.	Sets bitmap for acknowledge from Master. 0x00 will clear all flags.
Bind	'B' – 0x42	Register number (1-8) followed by 8 byte address (same order as for Destination/module address)	Used to bind slaves to master. Data stored in non-volatile memory.
Channel	'C' – 0x43	0x01-0x0C (1-10) Apply for R mode only	Data is stored in volatile memory only.
Decrypt	'D' – 0x44	One byte bitmap, address register 1 is LSB.	Sets bitmap for decryption of data. 0x00 will clear all flags.
Encrypt	'E' – 0x45	One byte bitmap, address register 1 is LSB.	Sets bitmap for encryption of data. 0x00 will clear all flags.
C-field	'F' – 0x46	0x00-0xFF (0-255)	Data is stored in volatile memory only.
M-Bus mode	'G' – 0x47	0x00-0x04 (0-4) 0: S 1: T1 2: T2 3: Reserved 4: R	Data is stored in volatile memory only. S and R mode not supported in NTA8130
Install	'I' – 0x49	0: Normal operation 1: Install mode 2: Accept all messages	In install mode messages with C-field = 06h and 46h are accepted. Use in Master only.
Key register	'K' – 0x4B	Slave: 16 byte key. Master: Register number (1-8) followed by 16 byte key	Used to set encryption key. Data stored in non-volatile memory.
Memory configuration	'M' – 0x4D	(Address, Data): see list of parameters below. 0xFF exits memory configuration.	Used to enter memory configuration menu. Parameters changed are stored in non-volatile memory.
Output power	'P' – 0x50	0x01-0x05 (1-5)	Data is stored in volatile memory only.
Quality Indicator	'Q' – 0x51	Returns one byte indicating the signal quality of the last received packet	Based on bit errors preamble and synch word
Signal Strength (RSSI)	'S' – 0x53	Returns one byte indicating the signal strength of a detected signal or a valid packet.	If a valid packet has been received when in configuration mode, it will return the RSSI of the last received packet.
Destination / module address	'T' – 0x54	8 bytes; M_ID2, M_ID1,	Data is stored in volatile memory only.

		U_ID4, U_ID3, U_ID2, U_ID1, VER (Version), DEV (Device Type),	
Exit command	'X' – 0x58	(none)	Exit to normal operation mode. All changes of parameters take effect.
Sleep mode	'Z' – 0x5A	(none)	Exit sleep mode by sending 0xFF on UART RXD pin
Test mode 0	'0' – 0x30	(none)	List all configuration memory parameters
Test mode 1	'1' – 0x31	(none)	TX carrier
Test mode 2	'2' – 0x32	(none)	TX modulated signal PN9 sequence
Test mode 3	'3' – 0x33	(none)	TX Off, RX mode

Note: ASCII characters are written as 'X', hexadecimal numbers are written like 0x00, and decimal numbers are written like 10 throughout the text. A table of ASCII characters and their respective hex and decimal values are found in the Appendix.

Any invalid command will be ignored and the '>' prompt will be re-sent.

If Test mode 1 or 2 is used, it is important to enter Test mode 3 before exiting the configuration mode ('X') in order to ensure proper operation in normal mode.

Example:
To select RF channel 3, send the follow sequence after asserting the CONFIG line and the '>' prompt is received:

Command	**Hex**	**Response**	**Comment/Note**
Enter	0x00	'>'	Or assert CONFIG pin De-assert CONFIG after '>' prompt
'C'	0x43	'>'	
3	0x03	'>'	Wait for '>' prompt
[A new command could be issued here]			
'X'	0x58	(none)	Module returns to IDLE state

Note that the CONFIG line must be de-asserted after the first '>' prompt was received, but before the 'X' command.

MBUS2 Configuration Memory

The table below shows the complete list of configurable parameters stored in non-volatile memory. These values can be changed using the 'M' command. All addresses and arguments must be sent as binary values to the module (not as ASCII representation for hex or decimal).

Parameter	Description	Address hex	Argument dec	Factory setting hex (dec)	Comment
Radio configuration					
RF_CHANNEL	Default RF channel for R mode only	0x00	1-10	0x01 (1)	See data sheet for channel frequencies. Only used for R mode.
RF_POWER	Default RF output power	0x01	1-5	0x05 (5)	See data sheet for output power levels.
MBUS_MODE	M-Bus mode	0x03	0-4 0: S 1: T1 2: T2 3: Reserved 4: R	0x01 (1)	Use 'G' command to change value in volatile memory only S and R mode not supported in NTA8130
SLEEP_MODE	Sleep mode	0x04	0: Disable Sleep 1: Enable Sleep 3: Enable with sleep timeout	0x00 (0)	When enabled the module enter Sleep mode after transmission. Delay set by RX_TIMEOUT. If enabled with sleep timeout, the module goes directly to Sleep after a Reset, and to Sleep after TIMEOUT when wakeup from Sleep or exit Config mode.
RSSI_MODE	Append RSSI to received data	0x05	0:Disabled 1: Enabled	0x00 (0)	When enabled the RSSI value is appended to the received data
TIMEOUT	Time before modem clear buffer without transmitting if Buffer size < Length byte (first byte).	0x10	<1-254> 0x01 (1): 32 ms 0x02 (2): 48 ms 0x03 (3): 64 ms 0x7C (124): 2 s 0xF9 (249): 4 s	0x7C	IF SLEEP_MODE=3 the same timeout is used to auto sleep slaves. Modem transmit without timeout when Buffer size = length byte.
Radio packet configuration					
PREAMBLE_ LENGTH	Short of long preamble in S mode only	0x0A	0x00 (0): Short 0x01 (1): Long	0x00 (0)	Preamble (header) length, apply for S mode only
Medium access, addressing and network management					
NETWORK_ROLE		0x12	0x00 (0): Slave/Meter 0x01 (1): Master/Concentrator	0x00 (0)	
M_ID1	Manufacturer ID, first byte	0x19	0x00-0xFF (0-255)	0x0C (12)	
M_ID2	Manufacturer ID, second byte	0x1A	0x00-0xFF (0-255)	0xAE (174)	
U_ID1	Unique ID, first byte	0x1B	0x00-0xFF (0-255)	0x12 (18)	
U_ID2	Unique ID, second byte	0x1C	0x00-0xFF (0-255)	0x34 (52)	
U_ID3	Unique ID, third byte	0x1D	0x00-0xFF (0-255)	0x56 (86)	
U_ID4	Unique ID, forth	0x1E	0x00-0xFF	0x78	

	byte		(0-255)	(120)	
VER	Version	0x1F	0x00-0xFF (0-255)	0x01 (1)	
DEV	Device	0x20	0x00-0xFF (0-255)	0x07 (7)	
Data and configuration interface, UART Serial Port					
UART_BAUD_RATE	Baud rate	0x30	0x00: Not used 0x01: 2400 0x02: 4800 0x03: 9600 0x04. 14400 0x05: 19200 0x06: 28800 0x07: 38400 0x08: 57600 0x09: 76800 0x0A: 115200 0x0B: 230400	0x05 (5)	BE CAREFUL IF CHANGING AS HOST MAY LOOSE CONTACT WITH MODULE! Does not take effect until module is re-booted / reset.
UART_FLOW_CTRL	UART flow control	0x35	0: None 1:CTS only 3:CTS/RTS 4:RXTX(RS485)	0x00 (0)	
DATA_INTERFACE	Data interface	0x36	0x00: MBUS packet with ID and address 0x01: Application data only 0x02: Reserved 0x03: Application data only with ack (00:3Eh) 0x04: Add start/stop byte 0x08: Add CRC 0x0C: Add start/stop byte and CRC	0x00 (0)	Sets receiver data format. First byte is always packet length (except when using start byte)
FREQ_CAL		0x39		Different for each module.	Found in factory and used by the module to minimise the total frequency tolerance. For firmware upgrade, read back the value and write it back after the upgrade.
LED_CONTROL		0x3A	0: Disabled 1: RX/TX indicator 2: UART/RF IDLE indicator	0x00 (0)	Use to enable LED0/LED1 for RX/TX packet indication or UART/RF IDLE mode indicator.
CONTROL_FIELD	C-field	0x3B	0x00-0xFF (0-255)	0x06 (6)	Use 'F' command to change value in volatile memory only
RX_TIMEOUT		0x3C	0x00-0xFF (0-255)	0x0B (11)	Delay before Sleep mode, n x 0.6 ms
INSTALL_MODE		0x3D	0: Normal mode (accept installed MBUS meters only) 1: Install mode 2: Filter off (accept all MBUS types)	2	
ENCRYPT_FLAG		0x3E		0	Bit mask for encryption, enabled when set
DECRYPT_FLAG		0x3F		0	Bit mask for decryption, enabled when set
DEFAULT_KEY		0x40-0x4F		All 0xFF (255)	
INIT_VECTOR		0x50-0x5F		All 0x00 (0)	

PART_NUMBER		0x60-0x6B		RCxxxx-MBUS2	
HW_REV_NO		0x6D-0x72		x.yz	x, y and z; Any number 0d-9d
FW_REV_NO		0x74-0x77		x.yz	x, y and z; Any number 0d-9d
ADDRESS_ID1		0x80-0x87		All 0x00	Address for installed meters.
ADDRESS_ID2		0x88-0x8F		All 0x00	Address for installed meters.
ADDRESS_ID3		0x90-0x97		All 0x00	Address for installed meters.
ADDRESS_ID4		0x98-0x9F		All 0x00	Address for installed meters.
ADDRESS_ID5		0xA0-0xA7		All 0x00	Address for installed meters.
ADDRESS_ID6		0xA8-0xAF		All 0x00	Address for installed meters.
ADDRESS_ID7		0xB0-0xB7		All 0x00	Address for installed meters.
ADDRESS_ID8		0xB8-0xBF		All 0x00	Address for installed meters.
SERIAL_NUMBER		0xC0-0xC7		All 0x00	8 bytes reserved for serial number for traceability. Is programmed by Radiocrafts during test.
Exit from memory configuration		0xFF	No argument should be sent		To exit from command mode the 'X' command must be sent after '>' is received.

To make permanent changes to default values and other parameters, the Memory Configuration command 'M' is used. This command should be followed by pairs of byte being the memory address and the new value to be stored at that address. In order to exit the Memory Configuration mode, the 'address' 0xFF must be sent, but without any data argument. Then wait for the '>' prompt while the internal memory is re-programmed (See Timing Information for typical delay). To completely exit from command mode, the normal exit command 'X' must be sent.

Example:
To change the MAN_ID (at address 0x19 and 0x1A) and set it to (100,200) (0x64,0xC8), send the following sequence:

Command	**Hex**	**Response**	**Comment/Note**
Enter	0x00	'>'	Or assert CONFIG pin De-assert CONFIG after '>' prompt
'M'	0x4D	'>'	Module ready to receive address
0x19	0x19	(none)	
100	0x64	(none)	
0x1A	0x1A	(none)	
200	0xC8	(none)	
[new address could be sent here]			
[new value could be sent here]			
0xFF	0xFF	'>'	Wait for '>' prompt
'X'	0x58	(none)	Module returns to IDLE state

Test mode 0 ('0' command) can be used to list all parameters stored in non-volatile memory. This command can be used to verify and check the module configuration.

MBUS3 Description

The OMS specification differs from NTA 8130 in the way the communication session between a slave (meter) and master (MUC) is done. For battery operated devices the slave is always initiate the communication, and the master must then transmit (if any) within a short time window (2-3 ms in the T mode). After one such “ping-pong” sequence, the slave will have a pause (enter sleep mode) for 2-5 seconds, before it again do a new transmission allowing the master to do another transmission. This means that all messages to be sent from the master must be ready and transmitted within a very short time. The MBUS3 Auto-message generator and Mailbox features make this task easy and doable.

The installation scenarios in OMS are supposed to support many meters per MUC, hence the address and registers has been extended to 64 positions in MBUS3.

MBUS3 Auto-message Generator
In T mode the Master must reply with a new message within 3 ms after a received Access Demand, if further communication shall take place. To meet this timing requirement, the module has built-in an Auto-message Generator and a Mailbox.

Auto-message Flags (set using the A –command) are used to indicate which slave shall be replied at the next access. The flag is automatically cleared when a reply is sent (when enabled), but can also be cleared manually with the A-command.

For each address register there is a corresponding flag register (two bytes):

SSSSSSSS, E:D:FCB:R:MB

Where

SSSSSSSS selects the standard message, see below
E is the Encryption flag bit
D is the Decryption flag bit
FCB is the Frame Count Bit (to be used in the next transmission)
R is one bit reserved for future use
MB is the Mailbox selected by 4 bits (0000 to 1111), where 0000 means none

The flags are set by using the A command: First send the register number, followed by the two bytes. The flags are stored in volatile memory and will be lost during power off.

The E and D flags can be set by default for all register positions using the ENCRYPT and DECRYPT configuration parameter.

The MAILBOX configuration parameter is used to set certain features of the Auto-message Generator:

RRR:ACSM:ACMB:AMMB:DFC:A

Where

RRR are three bits reserved for future use
ACSM is the Auto Clear Standard Message bit
ACMB is the Auto Clear Mailbox bit
AMMB is the Accept all Messages (C-fields) for Mailbox transmission bit
DFC is the check Data Flow Control bit in the C-field
A is the check Accessibility bit

If the ACSM or ACMB bit is set, the auto-message flag will be automatically cleared after a transmission. If the DFC bit is set, the DFC bit of the incoming message (in the C-field) will be checked before an auto-message is sent (otherwise it will be ignored, and the message sent regardless of the incoming message). If the A bit is set, the Accessability of the incoming message (in the Application Header signature) will be checked before an auto-message is

sent only to meters that signals they are accessible (otherwise it will be ignored, and the message sent regardless of the incoming message).

The Auto-message Generator supports three types of messages:

- Standard messages with fixed frame format
- Message from a mailbox
- A template messages from a mailbox

There are several standard messages that can be sent, depending on the incoming message from the slave. The master module will automatically recognize the type of message, and reply accordingly. The table below shows which messages can be sent for each incoming message, and the corresponding flags (SSSSSSSS) for the master reply. Standard messages shown in green cells, messages from the mailbox in red cells.

		Master reply						
		Mail-box	CNF-IR (0x06)	ACK (0x00)	REQ-UD1 (0x5A/ 7A*)	REQ-UD2 (0x5B/7B)	SND-NKE (0x40)	SND-UD (0x53/73)
Slave mess -age	SND-IR (0x46)		10000000					
	ACC-DMD (0x48)			01000000				
	SND-NR (0x44)				00000000: No reply 00100000: REQ-UD1 00010000: REQ-UD2 00001000: Reserved			←
	RSP-UD (0x08/28 0x18/38**)				00000000: No reply 00000100: REQ-UD1 00000010: REQ-UD2 00000001: SND-NKE			
	ACK (0x00/20 0x10/30**)							

*Note; the module will automatically set the Frame Count Bit in the C-field depending on the last transmitted FCB bit (as stored in the Flag Register). The FCB bit is automatically alternated.

** A reply is only sent if the DFC bit in the MAILBOX configuration parameter is cleared

The Status byte will automatically be set to the incoming packet RSSI value, and the Access Number will be set automatically, depending on the incoming message. Before a transmission, the Access Number can be set to a new value using the N-command. A reply to SND-IR and ACC-DMD use the incoming Access Number for the reply.

If several flags are set, the most significant bit flag has the highest priority. The Mailbox has priority above the standard messages. Mailbox messages will only be sent in reply to certain incoming message types as shown in the table above (in red). However, the module can also be configured (the AMMB parameter) to use the mailbox for any incoming message. The MAILBOX configuration parameter is used to set auto-clearing of flags for standard messages and the mailbox.

There are up to 15 mailboxes that can be used to store “pre-cooked” messages. Each mailbox is 64 bytes, except mailbox number 15 which is 128 bytes. Mailboxes can be combined to support messages up to 255 bytes (less the header). When using more than 64 bytes for one message, the following mailbox cannot be used. That is, if four mailboxes of 255 bytes are to be used, they should be address as number 1, 5, 9 and 13.

IMPORTANT NOTE: The binding (B-command) and Encryption Key entry (K-command) will erase the Mailbox (due to memory constraints in the module).

Mailboxes can be written and read using the W – write command, and the R – read command. The format for writing to the mailbox is:

Mailbox number, C-field, Length, CI-field, followed by the rest of the message

The Mailbox number should be 1-15. There is no restriction on the C-field value, but do note that OMS specify only a set of allowable C-fields. The Length byte shall be the number of bytes following, not including the Length byte itself. Note, if the message (including C and Length) is more than 64 bytes, the following mailbox cannot be used.

Long mailbox messages must be pre-encrypted in order to meet the time constraints in T mode. This is done by using the E-command. If the message holds a valid address, and the signature is set for encryption, and the Encryption flag is disabled, the message will be encrypted and stored in the Mailbox. Note, if the Encryption flag is not disabled, the message will be encrypted (again) when transmitted.

If the message in the mailbox has an Application Layer Address = 0, Access Number = 0 and Status = 0, the message is called a template. When using a template, the blank fields will be added on-the-fly when responding to the slave. This makes it possible to point at the same Template message for many meters.

The Access Number will automatically be increased for every transmission if set to zero. The RSSI of the last received packet will automatically be added in the Status byte by the module if set to zero.

Note; if the Template message requires encryption (as determined by the signature, and the Encryption enable flag, the time to encrypt the message on-the-fly might violate the 2-3 ms response time in T-mode. Templates with more than one block to be encrypted can only be used for S mode (up to 50 ms response time). In this case the pre-encrypted mailbox message must be used.

MBUS3 Automatic Addressing

When the Master receives a message from a slave, this slave's address will be used as the Application Layer Address for the next transmission from the master when using the auto-message feature.

When messages are sent from the UART buffer, the Application Layer Address used is determined by the host. Note; the Link Layer Address (in the MAC header) is always the masters own address as stored in the configuration memory. The Link Layer Address stored in configuration memory can be overridden by using the 'T' command (volatile memory), or permanently changed using the 'M' command (non-volatile memory).

MBUS3 Automatic Sleep

The Slave has special support for automatic sleep after data transmission. If automatic SLEEP after TX is enabled (SLEEP_MODE = 1), the module will automatically go to sleep in a configurable time after data transmission, if a message is not received. The receiver timeout is configured by RX_TIMEOUT. The automatic sleep is done after any message transmitted. The slave host application is responsible for waking up the module for a re-transmission in 2-5 seconds if a communication sequence was started, but no reply received from the master.

If automatic SLEEP after RX (and TX) is enabled (SLEEP_MODE = 3), the module will automatically go to sleep after data reception and the message is sent to the host over the UART. This feature should normally only be used for a Slave. In T-mode the RX_TIMEOUT should be set to 6. In S-mode the RX_TIMEOUT should be set to 86.

MBUS3 Installation and Binding

The module (Master) can be set in Installation Mode using the "I" command. When the module is in Installation Mode it will accept all Send Installation Request (SND-IR) messages (C-field is 46h).

Slaves can be bound to a Master by registering their addresses in the Address Register. This is done by using the "B" (Bind) command followed by a register number (1-64) and an 8 bytes slave address. The Slave address format is the same as used in the Application Layer Address. Thus, a maximum of 64 meters can be bound to one Master (MUC).

Note; the host must know which registers are used and which are free at any time.

MBUS3 Encryption

The module supports AES-128 encryption. When a slave is registered into the masters address register, the master host should request the encryption key from the utility data base. The new key is specific for each slave and related to the slave equipment ID or unique address.

Currently, OMS does not specify how to distribute new keys over the air to the slave.

The master host should send the new (plain) key to the master module using the 'K' command, followed by the register location number, and the 16 bytes (the new key). The register location number must correspond to the address register location for that slave.

The ENCRYPT_FLAG and DECRYPT_FLAG parameters are used to set the flags to enable / disable the encryption when transmitting and receiving messages.

For a message to be encrypted, the encryption flag for the particular slave must be set, and the CI-field and Signature field sent to the module must be according to the standard for encryption to take place. The message to be encrypted must have stuffing bytes ("2F") if required to get a full 16 byte encryption block. Only CI-fields 0x5A, 0x5B, 0x60, 0x61, 0x64, 0x65, 0x6C, 0x6D, 0x72, 0x7A, 0x7C, 0x7D, 0x7E, 0x7F, 0x80, 0x8A and 0x8B allows encryption. The Signature field must set encryption mode 0x04 or 0x05 according to OMS.

When using Signature field 0x05, the application must add the two encryption verification bytes (0x2F) after the header. The Initialization Vector for the encryption is extracted from the long header (for CI-fields 0x5B, 0x60, 0x64, 0x6C, 0x6D, 0x72, 0x7C, 0x7E, 0x80 and 0x8B). For the short header (CI-field fields 0x5A, 0x61, 0x65, 0x7A, 0x7D, 0x7F and 0x8A) the Initialization Vector is partly from the MAC header (destination address) and the short application header.

The Access Counter byte in the application header is used by the encryption, and the counter must be incremented in order to avoid repetitive messages. When using the mailbox template the counter is automatically incremented by the module.

For a message to be decrypted the decryption flag for the particular slave must be set, and the CI-field and Signature field must be according to the standard for encryption to take place.

In the slave, register 1 is used to hold the flags for encryption/decryption.

The set flags command 'A' is used to set decryption and encryption flags without storing in non-volatile memory.

The 'E' command can be used to encrypt a message in the Mailbox, as described above.

Note that you also have the option to use the M command to permanently set the Encryption/Decryption flag in the configuration non-volatile memory, instead of using the 'A' command. Encryption / decryption will only take place when the signature field indicate mode 0x04 or 0x05. If encryption mode 0x00 is used, the message will not be encrypted / decrypted even if the flags are set.

MBUS3 Repeater

The MBUS3 feature set contains complete autonomous unidirectional repeater functionality. The module can be configured as a Repeater by setting NETWORK_ROLE = 2. The MBUS_MODE must be set to 0 (S-mode) or 1 (T1 mode).

The Repeater operates as a stand-alone unit after power on, no installation is required. The Repeater should be mains powered as its receiver is working continuously. The Repeater can handle up to 15 messages simultaneously, each with a random delay before the transmission.

The Repeater will repeat all SND-NR and SND-IR messages within reach after a random delay of 5-25 seconds, with the hop counter set to 1. If it is a SND-IR message, the repeater will in addition generate a SND-NKE message after 2-5 seconds to indicate it is within range to support an installation tool. The SND-NKE message contains the RSSI level for the received SND-IR message.

The Repeater generates a SND-NR "management message" every 240 minutes. This message indicates that the Repeater is alive and working.

The Repeater can be set in Installation mode by activating (setting low) the Install button input (pin 25, Demo Board S5) for < 1 second. The LED driver output (pin 29, Demo Board D1, red LED) will blink rapidly when in Installation mode. In Installation mode the module transmits SND-IR every 30-60 seconds. After 90 transmissions it returns to normal mode.

The installation mode can also be turned off by activating the Install button input again. The LED output will then stop toggling.

All pending messages are cleared when entering Installation mode, but the module also works as a normal repeater when in Installation mode.

MBUS3 Timing table
The table below shows the timing information for the module when changing between different operating states. Timing symbol is according to figure 5 and 6.

Symbol	Value	Description / Note
t_{RX-TXD}	180 us	Time from last byte is received from the air until first character is sent on the UART
t_{TXD}	Min 590 us	t_{TXD} = # bytes received x 590 us/char (10 bits at 19.2 kBd + 70 us delay per character)
$t_{TXD-IDLE}$	900 us	Time from last character is sent on the UART until module is in IDLE mode (ready for RXD and RX)
$T_{RXD-CTS}$	20 us	Time from last character is received by the UART (including any timeout) until CTS is activated
t_{RXD-TX}	960 us	Time from last character is received by the UART (including any timeout) until the module sends the first byte on the air.
$T_{TX-IDLE}$	960 us	Time from last character is sent on the air until module is in IDLE mode (ready for RXD and RX)
$t_{OFF-IDLE}$	3.2 ms	
$t_{RESET-IDLE}$	3.0 ms	
$t_{SLEEP-IDLE}$	1.3 ms	
$t_{CONFIG-PROMPT}$	60 us	Time from 00h / CONFIG pin is set low until prompt (“>”)
$T_{G-CONFIG}$	1.1 ms	Delay after channel-byte is sent until prompt (“>”).(For other volatile memory commands there is no delay but immediate prompt)
$T_{G-CONFIG}$	1.1 ms	Delay after new M-Bus mode-byte is sent until prompt (“>”).(For other volatile memory commands there is no delay but immediate prompt)
T_{WAIT}	1.55 ms (B, K and M command) 24 us (all other commands)	Delay from stop bit of the command byte to start bit of the prompt reply. See figure 5 for details.
$T_{MEMORY-CONFIG}$	31 ms	In this period the internal flash (non-volatile memory) is programmed. *Do not reset, turn the module off, or allow any power supply dips in this period as it may cause permanent error in the Flash configuration memory. After the last command parameter byte the host should wait for the ‘>’ prompt before any further action is done to ensure correct re-configuration.*
$T_{CONFIG-IDLE}$	1.1 ms	End of ‘X’ to IDLE
t_{TX}	3.6 ms	TX time for T1 mode when Length=1 on the UART. Preamble, sync, CRC and MBUS address field added internally. Depends on M-Bus mode (T, S, R) and L

MBUS3 Configuration Commands

A list of commands is shown in the table below. Commands must be sent as ASCII characters or their corresponding binary value. All arguments must be sent as binary values to the module (not as ASCII representation for hex or decimal).

Parameter	Command	Argument in hex (decimal)	Note
Auto-message flags	'A' – 0x41	Register number (1-64) followed by 2 bytes	Sets auto-message flags for the Master and encryption flags.
Bind	'B' – 0x42	Register number (1-64) followed by 8 byte address (NB: Use application layer format, NOT the same order as for Destination/module address in link layer)	Used to bind slaves to master. Data stored in non-volatile memory.
Channel	'C' – 0x43	0x01-0x0C (1-10) Apply for R mode only	Data is stored in volatile memory only.
Encrypt	'E' – 0x45	Mailbox number (1-15)	
C-field	'F' – 0x46	0x00-0xFF (0-255)	Data is stored in volatile memory only.
M-Bus mode	'G' – 0x47	0x00-0x04 (0-4) 0: S2 1: T1 2: T2 3: S1 4: R	Data is stored in volatile memory only. R mode not supported in OMS
Install	'I' – 0x49	0: Normal operation 1: Install mode 2: Accept all messages	In install mode messages with C-field = 06h and 46h are accepted. Use in Master only.
Key register	'K' – 0x4B	Register number (1-64) followed by 16 byte key. Slave must use register 1.	Used to set encryption key. Data stored in non-volatile memory.
List binding	'L' – 0x4C	Register number (1-64)	Module responds with the address stored in the register (8 bytes)
Memory configuration	'M' – 0x4D	(Address, Data): see list of parameters below. 0xFF exits memory configuration.	Used to enter memory configuration menu. Parameters changed are stored in non-volatile memory.
Access Number	'N' – 0x4E	0x00 – 0xFF (0-255)	Set new Access Number
Read AutO-message flags	'O' – 0x4F	Register number (1-64)	Module responds with the auto-message flags (2 bytes)
Output power	'P' – 0x50	0x01-0x05 (1-5)	Data is stored in volatile memory only.
Quality Indicator	'Q' – 0x51	Returns one byte indicating the signal quality of the last received packet	Based on bit errors preamble and synch word
Read mailbox	'R'– 0x52	Register number (1-64). Will send the selected mailbox content to the UART.	

Signal Strength (RSSI)	'S' – 0x53	Returns one byte indicating the signal strength of a detected signal or a valid packet.	If a valid packet has been received when in configuration mode, it will return the RSSI of the last received packet.
Destination / module address	'T' – 0x54	8 bytes; M_ID2, M_ID1, U_ID4, U_ID3, U_ID2, U_ID1, VER (Version), DEV (Device Type),	Data is stored in volatile memory only.
Write mailbox	'W' – 0x57	Register number (1-64) followed by C, L, CI and the rest of the message.	
Exit command	'X' – 0x58	(none)	Exit to normal operation mode. All changes of parameters take effect.
Sleep mode	'Z' – 0x5A	(none)	Exit sleep mode by sending 0xFF on UART RXD pin
Test mode 0	'0' – 0x30	(none)	List all configuration memory parameters
Test mode 1	'1' – 0x31	(none)	TX carrier
Test mode 2	'2' – 0x32	(none)	TX modulated signal PN9 sequence
Test mode 3	'3' – 0x33	(none)	TX off, RX mode
Test mode 4	'4' – 0x34	(none)	IDLE (TX off, RX off)

Note: ASCII characters are written as 'X', hexadecimal numbers are written like 0x00, and decimal numbers are written like 10 throughout the text. A table of ASCII characters and their respective hex and decimal values are found in the Appendix.

Any invalid command will be ignored and the '>' prompt will be re-sent.

If Test mode 1 or 2 is used, it is important to enter Test mode 3 before exiting the configuration mode ('X') in order to ensure proper operation in normal mode.

Example:
To select RF channel 3, send the follow sequence after asserting the CONFIG line and the '>' prompt is received:

Command	**Hex**	**Response**	**Comment/Note**
Enter	0x00	'>'	Or assert CONFIG pin De-assert CONFIG after '>' prompt
'C'	0x43	'>'	
3	0x03	'>'	Wait for '>' prompt
[A new command could be issued here]			
'X'	0x58	(none)	Module returns to IDLE state

Note that the CONFIG line must be de-asserted after the first '>' prompt was received, but before the 'X' command.MBUS3 Configuration Memory

The table below shows the complete list of configurable parameters stored in non-volatile memory. These values can be changed using the 'M' command. All addresses and arguments must be sent as binary values to the module (not as ASCII representation for hex or decimal).

Parameter	Description	Address hex	Argument dec	Factory setting hex (dec)	Comment
Radio configuration					
RF_CHANNEL	Default RF channel for R mode only	0x00	1-10	0x01 (1)	See data sheet for channel frequencies. Only used for R mode.
RF_POWER	Default RF output power	0x01	1-5	0x05 (5)	See data sheet for output power levels.
MBUS_MODE	M-Bus mode	0x03	0-4 0: S2 1: T1 2: T2 3: S1 4: R	0x01 (1)	Use 'G' command to change value in volatile memory only R mode not supported in OMS
SLEEP_MODE	Sleep mode	0x04	0: Disable Sleep 1: Enable Sleep after TX 2: Reserved 3: Enable Sleep after TX and RX 5: As 1 with sleep timeout 5: As 3 with sleep timeout	0x00 (0)	When enabled the module enter Sleep mode after transmission (or reception). Delay set by RX_TIMEOUT If enabled with sleep timeout, the module goes directly to Sleep after a Reset, and to Sleep after TIMEOUT when wakeup from Sleep or exit Config mode.
RSSI_MODE	Append RSSI to received data	0x05	0:Disabled 1: Enabled	0x00 (0)	When enabled the RSSI value is appended to the received data
Radio packet configuration					
PREAMBLE_ LENGTH	Short of long preamble in S mode only	0x0A	0x00 (0): Short 0x01 (1): Long	0x00 (0)	Preamble (header) length, apply for S mode only
TIMEOUT	Time before modem clear buffer without transmitting if Buffer size < Length byte (first byte).	0x10	<1-254> 0x01 (1): 32 ms 0x02 (2): 48 ms 0x03 (3): 64 ms 0x7C (124): 2 s 0xF9 (249): 4 s	0x7C	IF SLEEP_MODE=3 the same timeout is used to auto sleep slaves. Modem transmit without timeout when Buffer size = length byte.
Medium access, addressing and network management					
NETWORK_ROLE		0x12	0x00 (0): Slave/Meter 0x01 (1): Master/Concentrator 0x10 (2): Repeater	0x00 (0)	
MAILBOX	Auto-message and mailbox configuration	0x16	0x10: Clear standard message flag 0x08: Clear mailbox flag 0x04: Use mailbox only for special messages* 0x02: Check		*Special messages are: SND-NR, RSP-UD and ACK

			DFC bit before transmission 0x01: Check Accessibility before transmission		
M_ID1	Manufacturer ID, first byte	0x19	0x00-0xFF (0-255)	0x0C (12)	
M_ID2	Manufacturer ID, second byte	0x1A	0x00-0xFF (0-255)	0xAE (174)	
U_ID1	Unique ID, first byte	0x1B	0x00-0xFF (0-255)	0x12 (18)	
U_ID2	Unique ID, second byte	0x1C	0x00-0xFF (0-255)	0x34 (52)	
U_ID3	Unique ID, third byte	0x1D	0x00-0xFF (0-255)	0x56 (86)	
U_ID4	Unique ID, forth byte	0x1E	0x00-0xFF (0-255)	0x78 (120)	
VER	Version	0x1F	0x00-0xFF (0-255)	0x01 (1)	
DEV	Device	0x20	0x00-0xFF (0-255)	0x07 (7)	
Data and configuration interface, UART Serial Port					
UART_BAUD_RATE	Baud rate	0x30	0x00: Not used 0x01: 2400 0x02: 4800 0x03: 9600 0x04. 14400 0x05: 19200 0x06: 28800 0x07: 38400 0x08: 57600 0x09: 76800 0x0A: 115200 0x0B: 230400	0x05 (5)	BE CAREFUL IF CHANGING AS HOST MAY LOOSE CONTACT WITH MODULE! Does not take effect until module is re-booted / reset.
UART_FLOW_CTRL	UART flow control	0x35	0: None 1:CTS only 3:CTS/RTS 4:RXTX(RS485)	0x00 (0)	
DATA_INTERFACE	Data interface	0x36	0x00: MBUS packet with ID and address 0x01: Application data only 0x02: Reserved 0x03: Application data only with ack (00:3Eh) 0x04: Add start/stop byte 0x08: Add CRC 0x0C: Add start/stop byte and CRC	0x00 (0)	Sets receiver data format. First byte is always packet length (except when using start byte)
FREQ_CAL		0x39		Different for each module.	Found in factory and used by the module to minimise the total frequency tolerance. For firmware upgrade, read back the value and write it back after the upgrade.
LED_CONTROL		0x3A	0: Disabled 1: RX/TX indicator 2: UART/RF IDLE indicator	0x00 (0)	Use to enable LED0/LED1 for RX/TX packet indication or UART/RF IDLE mode indicator.
CONTROL_FIELD	C-field	0x3B	0x00-0xFF (0-255)	0x06 (6)	Use 'F' command to change value in volatile

					memory only
RX_TIMEOUT		0x3C	0x00-0xFF (0-255) 0x06 (6) = 3 ms 0x56 (86) = 50 ms	0x06 (6)	Delay before Sleep mode, n x 0.6 ms Minimum 3 ms (6) in T mode, minimum 50 ms (86) in S mode.
INSTALL_MODE		0x3D	0: Normal mode (accept installed MBUS meters only) 1: Install mode 2: Filter off (accept all MBUS types)	2	
ENCRYPT_FLAG		0x3E	0: Disabled 1: Enabled	0	Default setting for encryption, enabled when set
DECRYPT_FLAG		0x3F	0: Disabled 1: Enabled	0	Default setting for decryption, enabled when set
DEFAULT_KEY		0x40-0x4F		All 0xFF (255)	
INIT_VECTOR		0x50-0x5F		All 0x00 (0)	
PART_NUMBER		0x60-0x6B		RCxxxx-MBUS3	
HW_REV_NO		0x6D-0x72		x.yz	x, y and z; Any number 0d-9d
FW_REV_NO		0x74-0x77		x.yz	x, y and z; Any number 0d-9d
SERIAL_NUMBER		0xC0-0xC7		All 0x00	8 bytes reserved for serial number for traceability. Is programmed by Radiocrafts during test.
Exit from memory configuration		0xFF	No argument should be sent		To exit from command mode the 'X' command must be sent after '>' is received.

To make permanent changes to default values and other parameters, the Memory Configuration command 'M' is used. This command should be followed by pairs of byte being the memory address and the new value to be stored at that address. In order to exit the Memory Configuration mode, the 'address' 0xFF must be sent, but without any data argument. Then wait for the '>' prompt while the internal memory is re-programmed (See Timing Information for typical delay). To completely exit from command mode, the normal exit command 'X' must be sent.

Example:
To change the MAN_ID (at address 0x19 and 0x1A) and set it to (100,200) (0x64,0xC8), send the following sequence:

Command	**Hex**	**Response**	**Comment/Note**
Enter	0x00	'>'	Or assert CONFIG pin De-assert CONFIG after '>' prompt
'M'	0x4D	'>'	Module ready to receive address
0x19	0x19	(none)	
100	0x64	(none)	
0x1A	0x1A	(none)	
200	0xC8	(none)	
[new address could be sent here]			
[new value could be sent here]			
0xFF	0xFF	'>'	Wait for '>' prompt
'X'	0x58	(none)	Module returns to IDLE state

Test mode 0 ('0' command) can be used to list all parameters stored in non-volatile memory. This command can be used to verify and check the module configuration.

Appendix: MBUS Command list overview

Command list	Feature set		
	MBUS1	MBUS2	MBUS3
'A' – 0x41	N.A	Acknowledge	N.A (controlled in flag register)
'B' – 0x42	N.A	Bind	Bind
'C' – 0x43	Channel	Channel	Channel
'D' – 0x44	N.A	Decrypt RF message before send to UART	N.A (controlled in flag register)
'E' – 0x45	N.A	Encrypt UART message before RF transmit	Encrypt mailbox message
'F' – 0x46	C-field	C-field	C-field
'G' – 0x47	M-Bus mode	M-Bus mode	M-Bus mode
'I' – 0x49	N.A	Install	Install
'K' – 0x4B	N.A	Key register	Key register
'L' - +x4C	N.A	N.A	List Binding
'M' – 0x4D	Memory configuration	Memory configuration	Memory configuration
'N' – 0x4E	N.A	N.A	Access number
'O' – 0x4F	N.A	N.A	Read Auto message flag register
'P' – 0x50	Output power	Output power	Output power
'Q' – 0x51	Quality Indicator	Quality Indicator	Quality Indicator
'R – 0x52'	N.A	N.A	Read Mailbox
'S' – 0x53	Signal Strength (RSSI)	Signal Strength (RSSI)	Signal Strength (RSSI)
'T' – 0x54	Destination address	Destination address	Destination address
'W' – 0x57	N.A	N.A	Write Mailbox
'X' – 0x58	Exit command	Exit command	Exit command
'Z' – 0x5A	Sleep mode	Sleep mode	Sleep mode
'0' – 0x30	Test mode 0	Test mode 0	Test mode 0
'1' – 0x31	Test mode 1	Test mode 1	Test mode 1
'2' – 0x32	Test mode 2	Test mode 2	Test mode 2
'3' – 0x33	Test mode 3	Test mode 3	Test mode 3
'4' – 0x34	Test mode 4	Test mode 4	Test mode 4

Commands in grey are stored in non-volatile memory. The rest of the commands are stored in volatile memory and is lost after a power off or a reset.

Appendix: Configuration Memory Factory Default

Address	MBUS1 factory default Values							
0x00-0x07	0x01	0x05	0x03	0x01	0x00	0x00	0x64	0x00
0x08-0x0F	0x00	0x00	0x00	0xD3	0x91	0xDA	0x80	0x80
0x10-0x17	0x7C	0x00	0x00	0x01	0x00	0x00	0x00	0x00
0x18-0x1F	0x00	0x0C	0xAE	0x12	0x34	0x56	0x78	0x01
0x20-0x27	0x07	0x01	0x01	0x00	0x00	0x00	0x00	0x04
0x28-0x2F	0xFF	0x08	0x00	0x00	0x00	0x00	0x00	0x00
0x30-0x37	0x05	0x08	0x00	0x01	0x05	0x00	0x00	0x01
0x38-0x3F	0x2B	0x00	0x01	0x44	0x00	0x52	0x43	0x31
0x40-0x47	0x31	0x38	0x30	0x2D	0x4D	0x42	0x55	0x53
0x48-0x4F	0x2C	0x31	0x2E	0x30	0x30	0x2C	0x31	0x2E
0x50-0x57	0x31	0x35	0x00	0xFF	0xFF	0xFF	0xFF	0xFF
0x58-0x5F	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0x60-0x67	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0x68-0x6F	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0x70-0x77	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0x78-0x7F	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
Address	**MBUS2 factory default Values**							
0x00-0x07	0x01	0x05	0x03	0x02	0x00	0x00	0x64	0x00
0x08-0x0F	0x00	0x00	0x00	0xD3	0x91	0xDA	0x80	0x80
0x10-0x17	0x7C	0x00	0x01	0x01	0x00	0x00	0x00	0x00
0x18-0x1F	0x00	0x0C	0xAE	0x12	0x34	0x56	0x78	0x01
0x20-0x27	0x07	0x01	0x01	0x00	0x00	0x00	0x00	0x04
0x28-0x2F	0xFF	0x08	0x00	0x00	0x00	0x00	0x00	0x00
0x30-0x37	0x05	0x08	0x00	0x01	0x05	0x00	0x00	0x01
0x38-0x3F	0x2B	0x00	0x00	0x06	0x0B	0x02	0x00	0x00
0x40-0x47	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0x48-0x4F	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0x50-0x57	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
0x58-0x5F	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
0x60-0x67	0x00	0x52	0x43	0x31	0x31	0x38	0x30	0x2D
0x68-0x6F	0x4D	0x42	0x55	0x53	0x2C	0x31	0x2E	0x30
0x70-0x77	0x30	0x2C	0x32	0x2E	0x30	0x30	0x00	0xFF
0x78-0x7F	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0x80-0x87	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
0x88-0x8F	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
0x90-0x97	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
0x98-0x9F	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
0xA0-0xA7	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
0xA8-0xAF	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
0xB0-0xB7	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
0xB8-0xBF	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
0xC0-0xC7	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0xC8-0xCF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0xD0-0xD7	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0xD8-0xDF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0xE0-0xE7	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0xE8-0xEF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0xF0-0xF7	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0xF8-0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF

Address	MBUS3 factory default Values							
0x00-0x07	0x01	0x05	0x03	0x01	0x00	0x00	0x64	0x00
0x08-0x0F	0x05	0x3C	0x00	0xD3	0x91	0xDA	0x80	0x80
0x10-0x17	0x7C	0x00	0x01	0x01	0x00	0x00	0x17	0x00
0x18-0x1F	0x00	0x0C	0xAE	0x12	0x34	0x56	0x78	0x01
0x20-0x27	0x07	0x01	0x01	0x00	0x00	0x00	0x00	0x04
0x28-0x2F	0xFF	0x08	0x00	0x00	0x00	0x00	0x00	0x00
0x30-0x37	0x05	0x08	0x00	0x01	0x05	0x00	0x00	0x01
0x38-0x3F	0x2B	0x00	0x00	0x06	0x06	0x02	0x00	0x00
0x40-0x47	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0x48-0x4F	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0x50-0x57	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
0x58-0x5F	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
0x60-0x67	0x00	0x52	0x43	0x31	0x31	0x38	0x30	0x2D
0x68-0x6F	0x4D	0x42	0x55	0x53	0x33	0x2C	0x32	0x2E
0x70-0x77	0x30	0x30	0x2C	0x33	0x2E	0x30	0x30	0x00
0x78-0x7F	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0x80-0x87	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0x88-0x8F	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0x90-0x97	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0x98-0x9F	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0xA0-0xA7	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0xA8-0xAF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0xB0-0xB7	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0xB8-0xBF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0xC0-0xC7	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0xC8-0xCF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0xD0-0xD7	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0xD8-0xDF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0xE0-0xE7	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0xE8-0xEF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0xF0-0xF7	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0xF8-0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF

Grey: Reserved (do not change).
Blue: Reserved for Module part number and version information (do not change). Typical values shown, the actual content of memory location 60d-83d depends on part number and version number

Appendix: ASCII Table

HEX	DEC	CHR	CTRL
0	0	NUL	^@
1	1	SOH	^A
2	2	STX	^B
3	3	ETX	^C
4	4	EOT	^D
5	5	ENQ	^E
6	6	ACK	^F
7	7	BEL	^G
8	8	BS	^H
9	9	HT	^I
0A	10	LF	^J
0B	11	VT	^K
0C	12	FF	^L
0D	13	CR	^M
0E	14	SO	^N
0F	15	SI	^O
10	16	DLE	^P
11	17	DC1	^Q
12	18	DC2	^R
13	19	DC3	^S
14	20	DC4	^T
15	21	NAK	^U
16	22	SYN	^V
17	23	ETB	^W
18	24	CAN	^X
19	25	EM	^Y
1A	26	SUB	^Z
1B	27	ESC	
1C	28	FS	
1D	29	GS	
1E	30	RS	
1F	31	US	
20	32	SP	
21	33	!	
22	34	"	
23	35	#	
24	36	$	
25	37	%	
26	38	&	
27	39	'	
28	40	(	
29	41	)	
2A	42	*	
2B	43	+	
2C	44	,	
2D	45	–	
2E	46	.	
2F	47	/	
30	48	0	
31	49	1	
32	50	2	
33	51	3	
34	52	4	
35	53	5	
36	54	6	
37	55	7	
38	56	8	
39	57	9	
3A	58	:	
3B	59	;	
3C	60	<	
3D	61	=	
3E	62	>	
3F	63	?	

HEX	DEC	CHR
40	64	@
41	65	A
42	66	B
43	67	C
44	68	D
45	69	E
46	70	F
47	71	G
48	72	H
49	73	I
4A	74	J
4B	75	K
4C	76	L
4D	77	M
4E	78	N
4F	79	O
50	80	P
51	81	Q
52	82	R
53	83	S
54	84	T
55	85	U
56	86	V
57	87	W
58	88	X
59	89	Y
5A	90	Z
5B	91	[
5C	92	\
5D	93	]
5E	94	^
5F	95	_
60	96	`
61	97	a
62	98	b
63	99	c
64	100	d
65	101	e
66	102	f
67	103	g
68	104	h
69	105	i
6A	106	j
6B	107	k
6C	108	l
6D	109	m
6E	110	n
6F	111	o
70	112	p
71	113	q
72	114	r
73	115	s
74	116	t
75	117	u
76	118	v
77	119	w
78	120	x
79	121	y
7A	122	z
7B	123	{
7C	124	|
7D	125	}
7E	126	~
7F	127	DEL

Document Revision History

Document Revision	Changes
1.0	First release
1.10	Detailed UART interface and Encryption example included. Minor changes and corrections
1.11	Timing and figure 3 corrections. Auto sleep included. Minor corrections in text.
1.20	Added MBUS3
1.21	Clarified order of address bytes for binding.
1.22	MBUS3 updates before official release.
1.30	MBUS2 new features for FW 2.20: - Auto sleep functionality update for slaves. - 8 Byte serial number reservations in configuration memory. - 1 byte frequency tolerance calibration value in configuration memory.
1.31	- Changed terminology from RC1180-MBUSx to MBUSx - Included quick start chapter (and removed it from data sheet 2.20) - RSSI reading included (and removed it from data sheet rev 2.20) - Optional custom specific version chapter included for MBUS2 - SLEEP_MODE, TIMEOUT and LED CONTROL update for MBUS2
1.40	MBUS3 new features for FW 3.09: - 8 Byte serial number reservations in configuration memory. - 1 byte frequency tolerance calibration value in configuration memory. - -SLEEP_MODE, TIMEOUT and LED CONTROL update

Disclaimer

Radiocrafts AS believes the information contained herein is correct and accurate at the time of this printing. However, Radiocrafts AS reserves the right to make changes to this product without notice. Radiocrafts AS does not assume any responsibility for the use of the described product; neither does it convey any license under its patent rights, or the rights of others. The latest updates are available at the Radiocrafts website or by contacting Radiocrafts directly. As far as possible, major changes of product specifications and functionality, will be stated in product specific Errata Notes published at the Radiocrafts website. Customers are encouraged to check regularly for the most recent updates on products and support tools.

Trademarks

RC232™ is a trademark of Radiocrafts AS. The RC232™ Embedded RF Protocol is used in a range of products from Radiocrafts. The protocol handles host communication, data buffering, error check, addressing and broadcasting. It supports point-to-point, point-to-multipoint and peer-to-peer network topologies.
All other trademarks, registered trademarks and product names are the sole property of their respective owners.

Life Support Policy

This Radiocrafts product is not designed for use in life support appliances, devices, or other systems where malfunction can reasonably be expected to result in significant personal injury to the user, or as a critical component in any life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness. Radiocrafts AS customers using or selling these products for use in such applications do so at their own risk and agree to fully indemnify Radiocrafts AS for any damages resulting from any improper use or sale.

© 2010, Radiocrafts AS. All rights reserved.

Contact Information

Web site: www.radiocrafts.com

Address:
Radiocrafts AS
Sandakerveien 64
NO-0484 OSLO
NORWAY

Tel: +47 4000 5195
Fax: +47 22 71 29 15
E-mails: radiocrafts@radiocrafts.com
sales@radiocrafts.com
support@radiocrafts.com